Computer Security Project

Cipher Craft Website

Submitted By:

Mohamed Ali Mohamed Ali Ragab

Supervised By:

Prof. Dr. Mahmoud E. Elshishtawy

T.A: AbdelRahman Essam

T.A: Razan Allaa

Table of Contents

1. Overview 3

2. Features 8

2.1. Cryptographic Algorithms 8

2.2. User Interface 8

2.3. Interactivity 8

2.4. Accessibility 8

3. System Requirements 9

4. Usage 9

4.1. Accessing the Dashboard 9

4.2. Encrypting/Decrypting Text 9

4.3. Exploring Examples 9

4.4. Switching Tabs 9

4.5. Toggling Dark Mode 10

5. Technical Details 10

5.1. Frontend 10

5.2. Backend 10

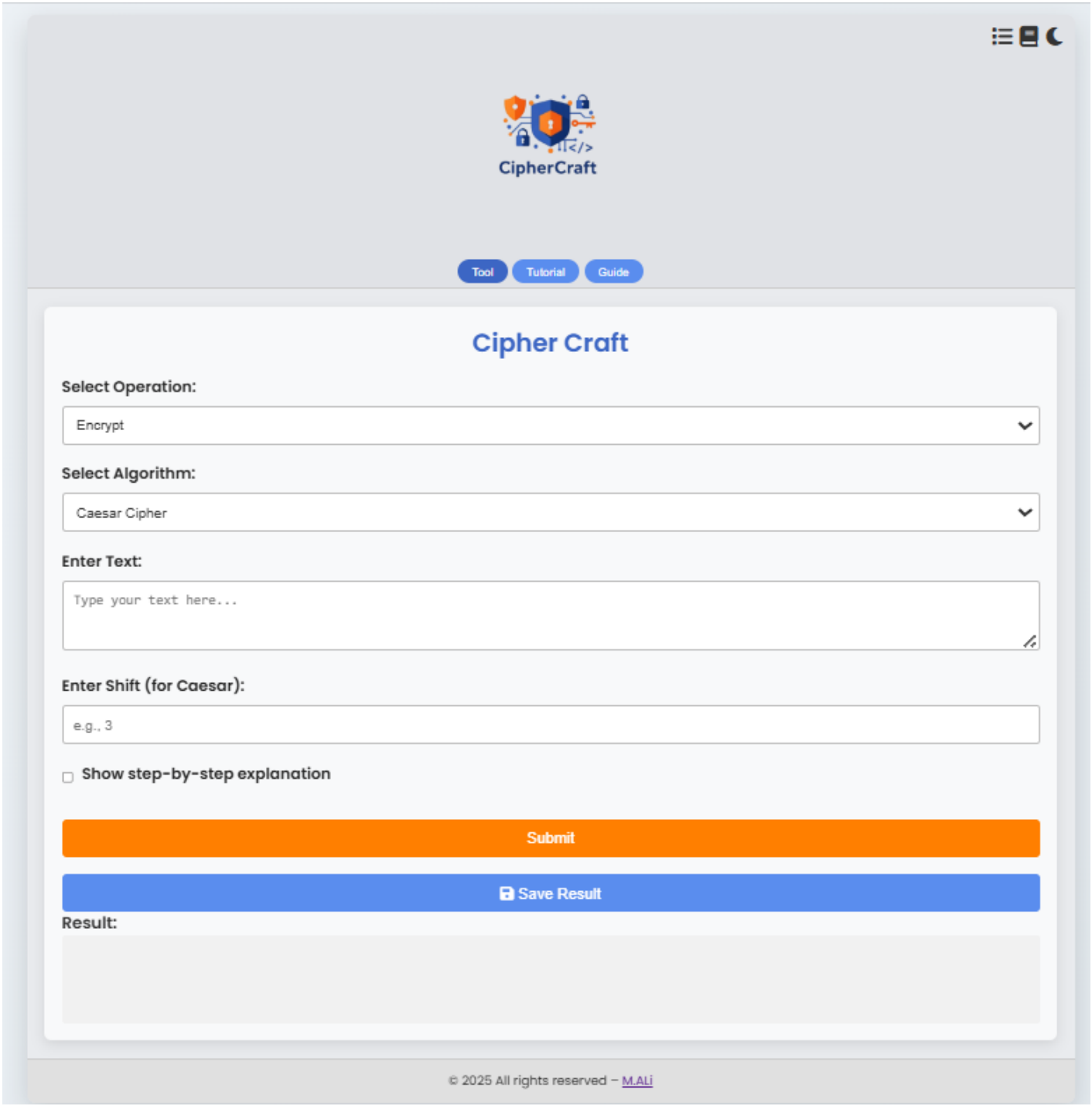
5.3. Dependencies 10

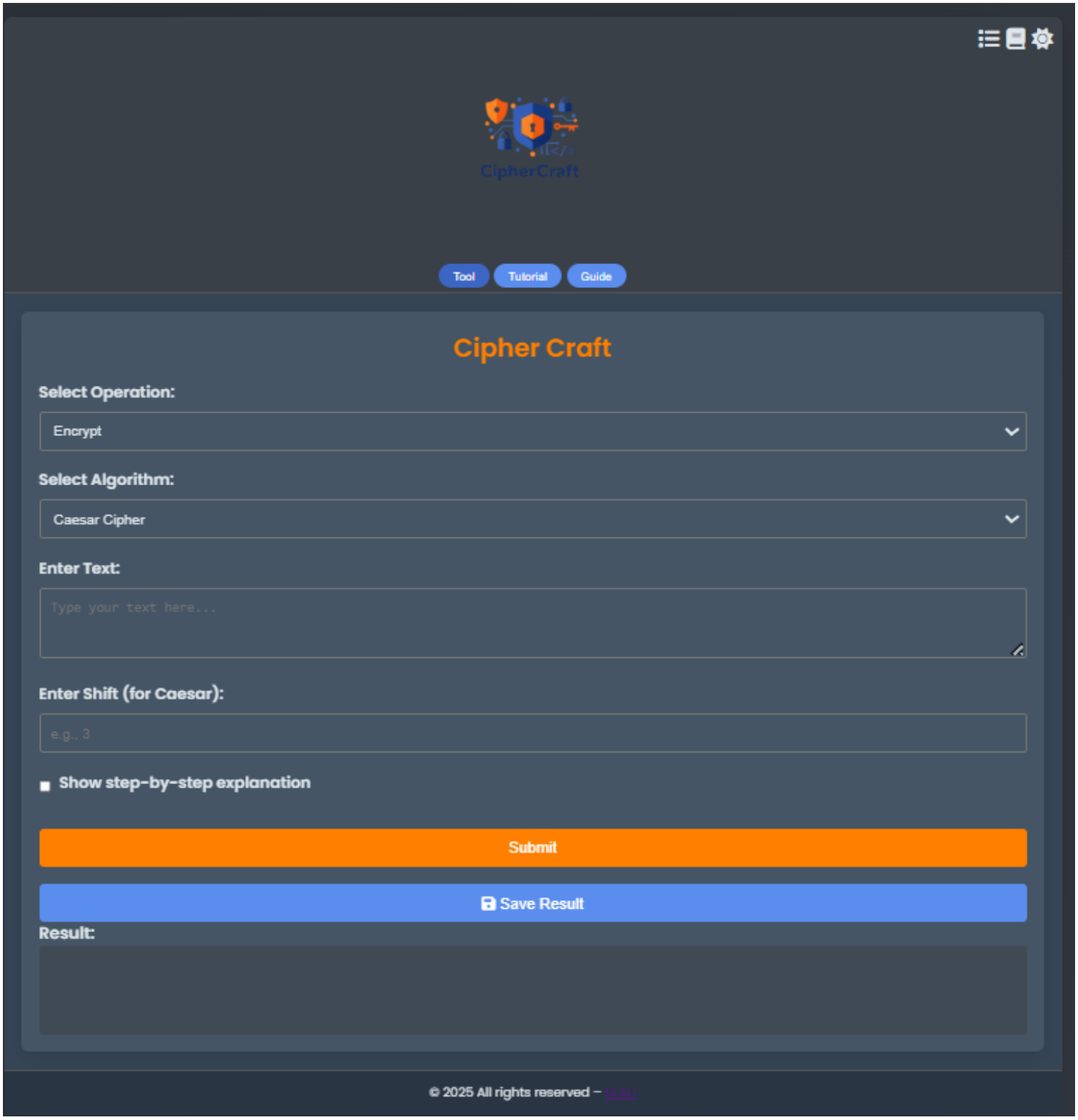
6. Example Usage 11

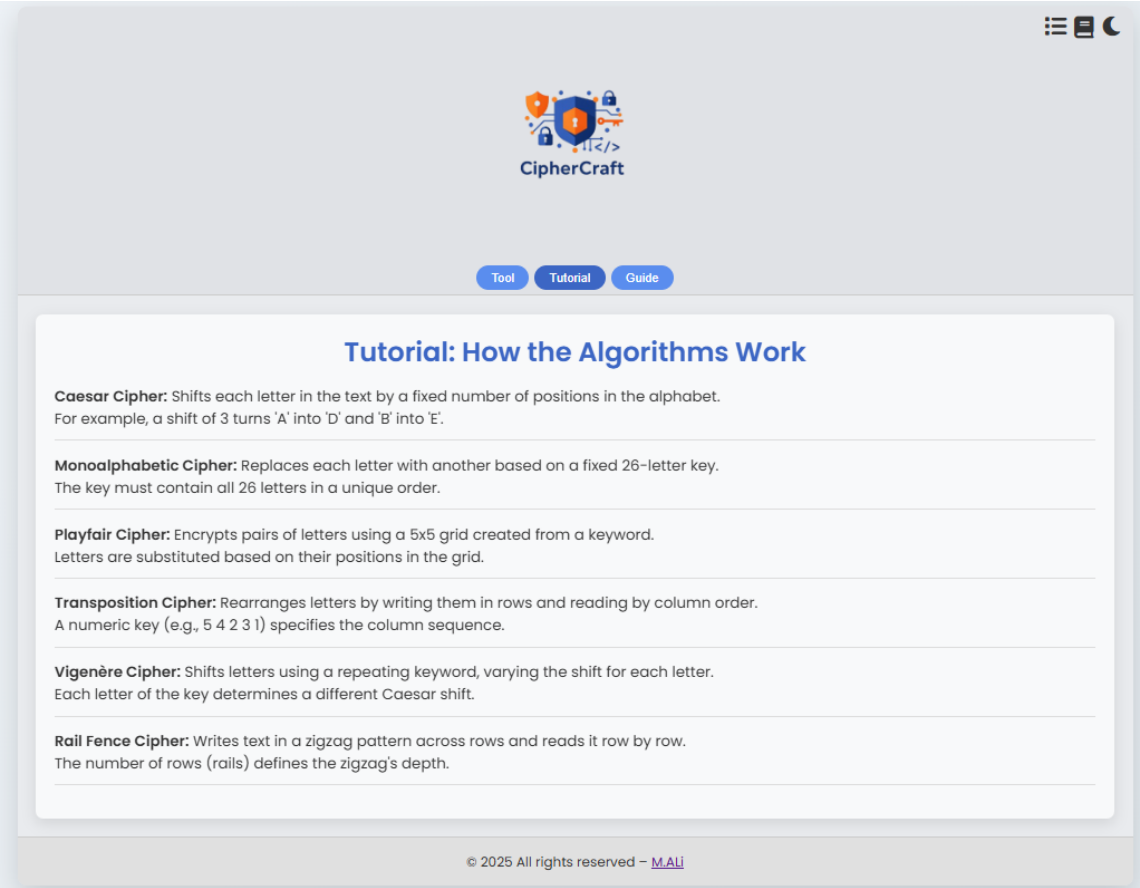
7. Resources 11

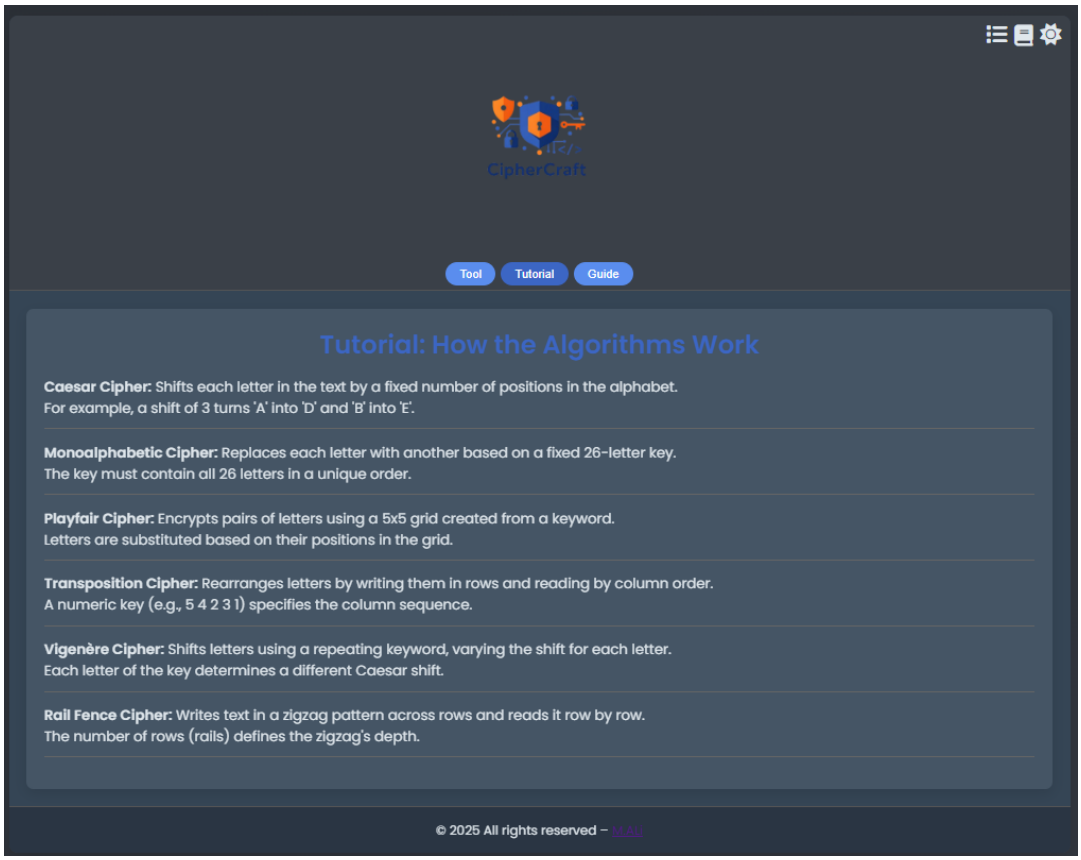
1. Overview

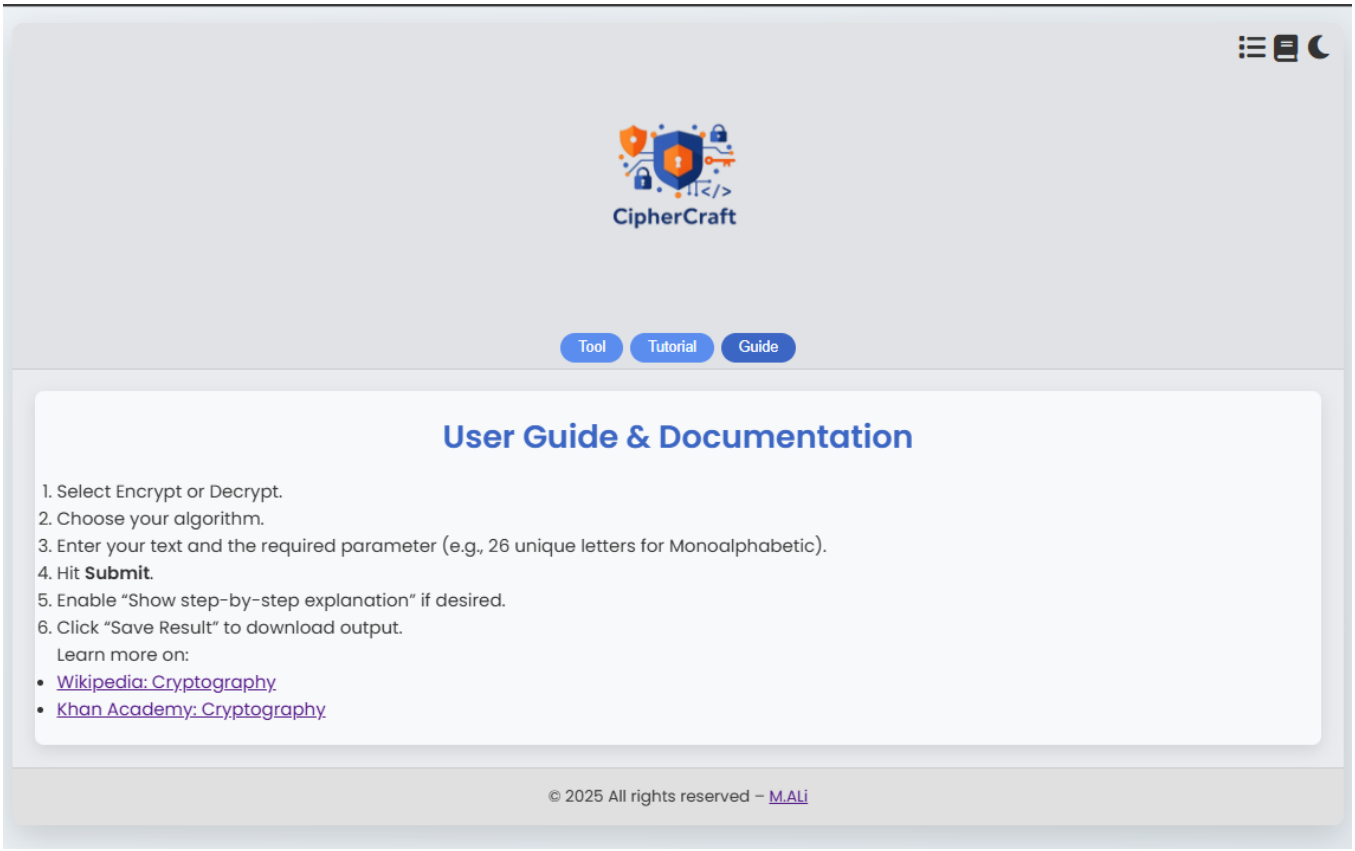
CipherCraft Dashboard is a web-based application designed to perform encryption and decryption using various classical cryptographic algorithms. Built with a user-friendly interface and robust backend, it serves as an educational tool for learning about cryptography and a practical utility for applying ciphers. The project supports **six ciphers:** Caesar, Monoalphabetic, Playfair, Transposition, Vigenère, and Rail Fence.

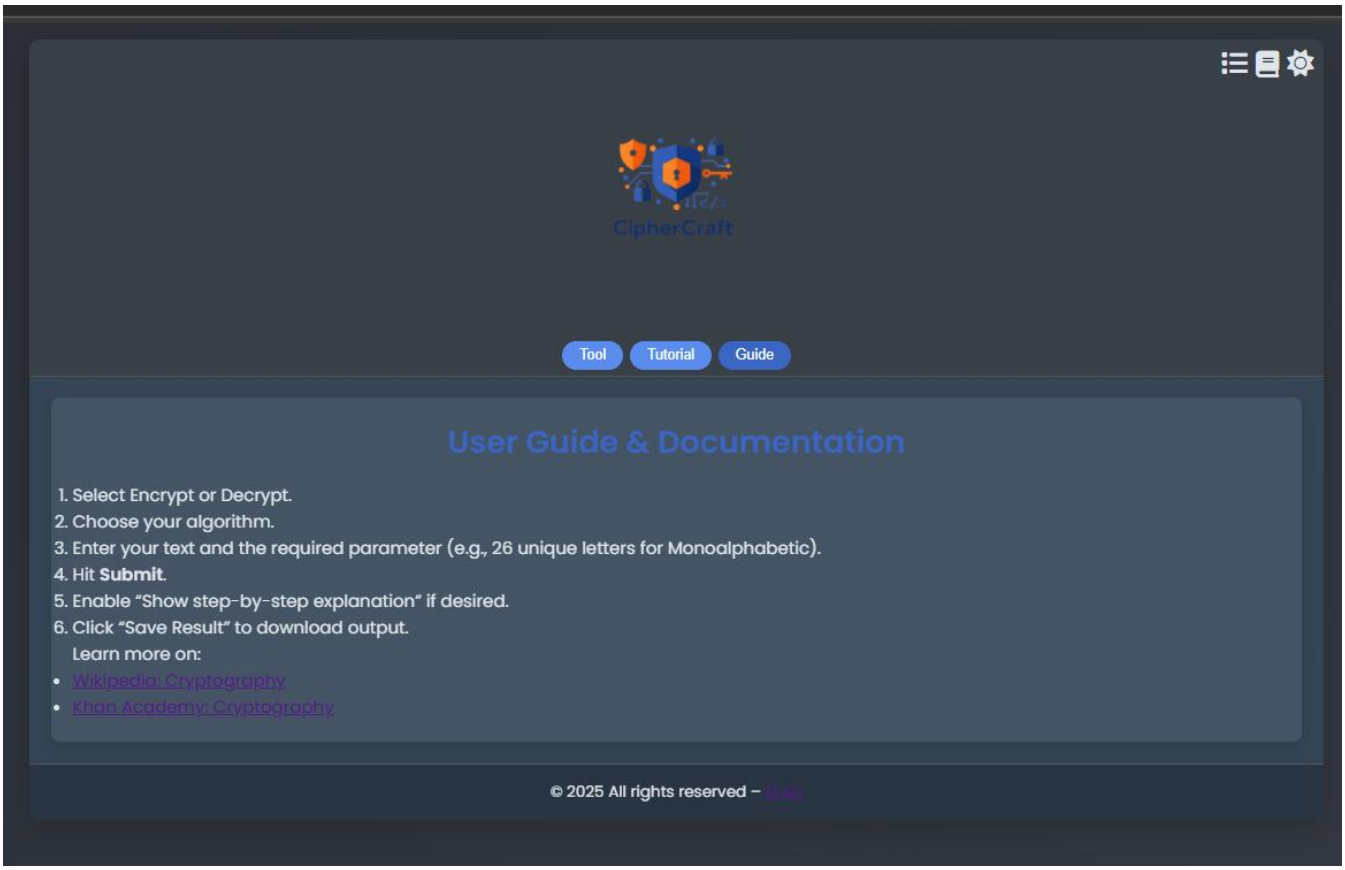


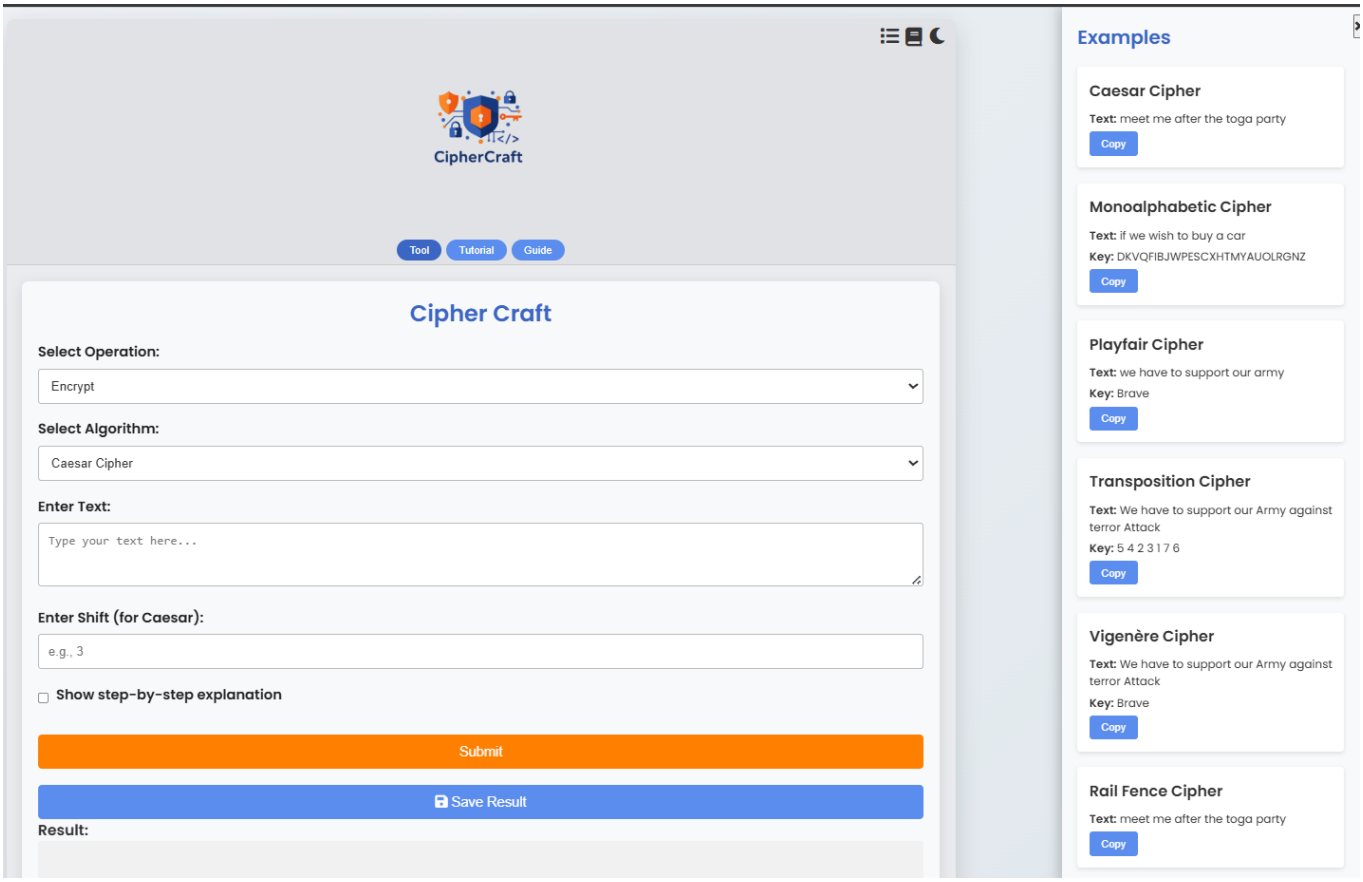

CipherCraft
Tool
Tutorial
Guide
Cipher Craft
Select Operation:
Encrypt
Select Algorithm:
Caesar Cipher
Enter Text:
Type your text here...
Enter Shift (for Caesar):
e.g., 3
Show step-by-step explanation
Submit
Save Result
Result:
© 2025 All rights reserved – MLAU

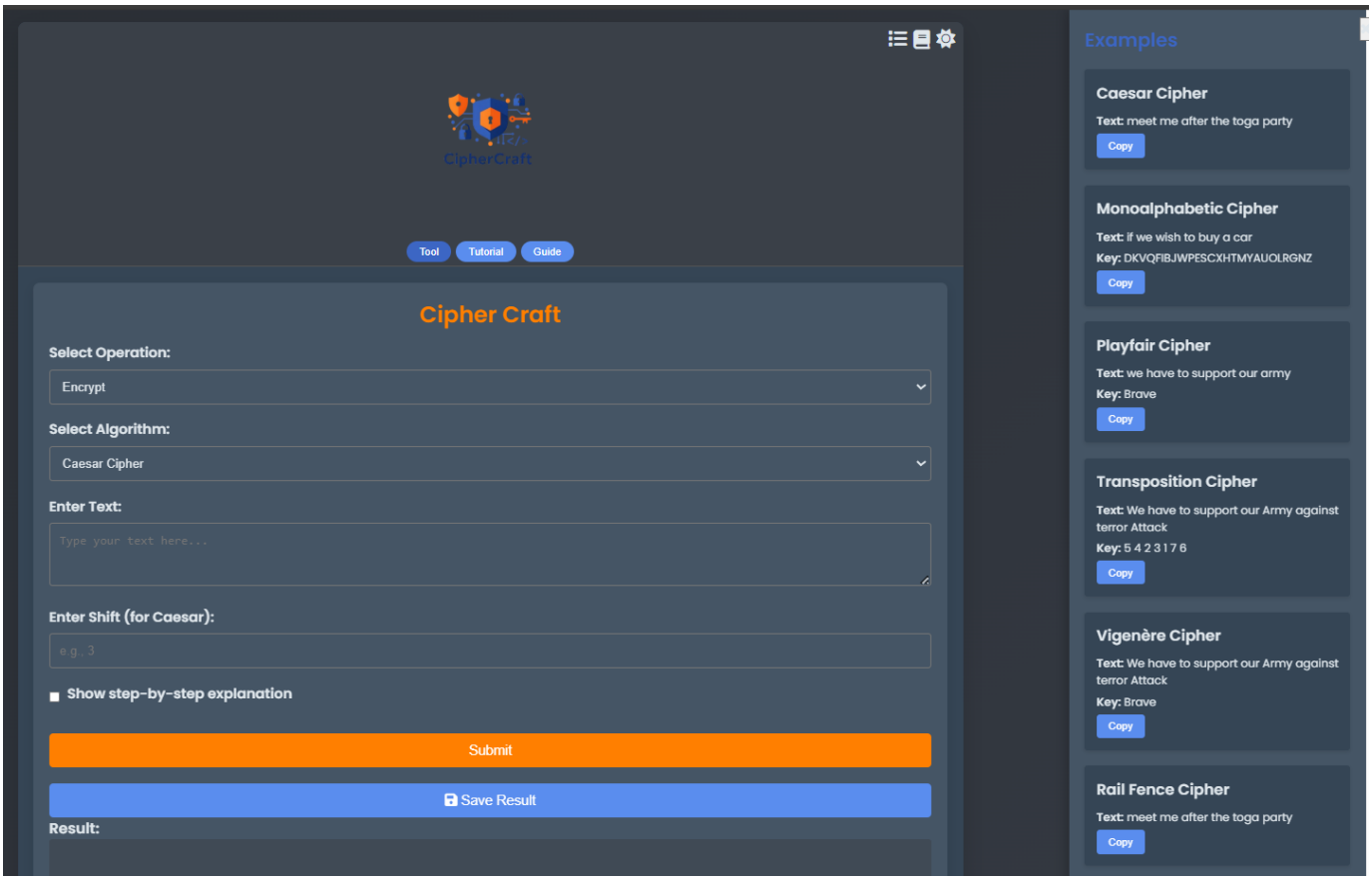

CipherCraft
Tool
Tutorial
Guide
Tutorial: How the Algorithms Work
Caesar Cipher: Shifts each letter in the text by a fixed number of positions in the alphabet.
For example, a shift of 3 turns 'A' into 'D' and 'B' into 'E'.
Monoalphabetic Cipher: Replaces each letter with another based on a fixed 26-letter key.
The key must contain all 26 letters in a unique order.
Playfair Cipher: Encrypts pairs of letters using a 5x5 grid created from a keyword.
Letters are substituted based on their positions in the grid.
Transposition Cipher: Rearranges letters by writing them in rows and reading by column order.
A numeric key (e.g., 5 4 2 3 1) specifies the column sequence.
Vigenère Cipher: Shifts letters using a repeating keyword, varying the shift for each letter.
Each letter of the key determines a different Caesar shift.
Rail Fence Cipher: Writes text in a zigzag pattern across rows and reads it row by row.
The number of rows (rails) defines the zigzag's depth.
© 2025 All rights reserved – M.ALI


CipherCraft
Tool
Tutorial
Guide
Tutorial: How the Algorithms Work
Caesar Cipher: Shifts each letter in the text by a fixed number of positions in the alphabet.
For example, a shift of 3 turns 'A' into 'D' and 'B' into 'E'.
Monoalphabetic Cipher: Replaces each letter with another based on a fixed 26-letter key.
The key must contain all 26 letters in a unique order.
Playfair Cipher: Encrypts pairs of letters using a 5x5 grid created from a keyword.
Letters are substituted based on their positions in the grid.
Transposition Cipher: Rearranges letters by writing them in rows and reading by column order.
A numeric key (e.g., 5 4 2 3 1) specifies the column sequence.
Vigenère Cipher: Shifts letters using a repeating keyword, varying the shift for each letter.
Each letter of the key determines a different Caesar shift.
Rail Fence Cipher: Writes text in a zigzag pattern across rows and reads it row by row.
The number of rows (rails) defines the zigzag's depth.
© 2025 All rights reserved – M.ALI


CipherCraft
Tool
Tutorial
Guide
User Guide & Documentation
1. Select Encrypt or Decrypt.
2. Choose your algorithm.
3. Enter your text and the required parameter (e.g., 26 unique letters for Monoalphabetic).
4. Hit Submit.
5. Enable "Show step-by-step explanation" if desired.
6. Click "Save Result" to download output.
Learn more on:
Wikipedia: Cryptography
Khan Academy: Cryptography
© 2025 All rights reserved – M.ALi


CipherCraft
Tool
Tutorial
Guide
User Guide & Documentation
1. Select Encrypt or Decrypt.
2. Choose your algorithm.
3. Enter your text and the required parameter (e.g., 26 unique letters for Monoalphabetic).
4. Hit Submit.
5. Enable "Show step-by-step explanation" if desired.
6. Click "Save Result" to download output.
Learn more on:
Wikipedia: Cryptography
Khan Academy: Cryptography
© 2025 All rights reserved –


CipherCraft
Tool
Tutorial
Guide
Cipher Craft
Select Operation:
Encrypt
Select Algorithm:
Caesar Cipher
Enter Text:
Type your text here...
Enter Shift (for Caesar):
e.g., 3
Show step-by-step explanation
Submit
Save Result
Result:
Examples
Caesar Cipher
Text: meet me after the toga party
Copy
Monoalphabetic Cipher
Text: if we wish to buy a car
Key: DKVQFIBJWPESCXHTMYAUOLRGNZ
Copy
Playfair Cipher
Text: we have to support our army
Key: Brave
Copy
Transposition Cipher
Text: We have to support our Army against terror Attack
Key: 5 4 2 3 1 7 6
Copy
Vigenère Cipher
Text: We have to support our Army against terror Attack
Key: Brave
Copy
Rail Fence Cipher
Text: meet me after the toga party
Copy


CipherCraft
Tool
Tutorial
Guide
Cipher Craft
Select Operation:
Encrypt
Select Algorithm:
Caesar Cipher
Enter Text:
Type your text here...
Enter Shift (for Caesar):
e.g., 3
Show step-by-step explanation
Submit
Save Result
Result:
Examples
Caesar Cipher
Text: meet me after the toga party
Copy
Monoalphabetic Cipher
Text: if we wish to buy a car
Key: DKVQFIBJWPESCXHTMYAUOLRGNZ
Copy
Playfair Cipher
Text: we have to support our army
Key: Brave
Copy
Transposition Cipher
Text: We have to support our Army against terror Attack
Key: 5 4 2 3 1 7 6
Copy
Vigenère Cipher
Text: We have to support our Army against terror Attack
Key: Brave
Copy
Rail Fence Cipher
Text: meet me after the toga party
Copy

2. Features

2.1. Cryptographic Algorithms

1. **Caesar Cipher**: Shifts letters by a fixed number of positions.
2. **Monoalphabetic Cipher**: Substitutes letters using a 26-letter key.
3. **Playfair Cipher**: Encrypts letter pairs using a 5x5 grid.
4. **Transposition Cipher**: Rearranges letters based on a numeric key.
5. **Vigenère Cipher**: Uses a keyword for polyalphabetic substitution.
6. **Rail Fence Cipher**: Writes text in a zigzag pattern across rails.

2.2. User Interface

- Tabbed navigation for Tool, Tutorial, and Guide sections.
- Responsive design with light and dark modes (persistent via localStorage).
- Examples panel with pre-filled inputs for quick testing.
- Step-by-step explanation option for algorithm processes.
- Save result feature to download outputs as text files.

2.3. Interactivity

- Dynamic parameter input based on selected algorithm (e.g., shift for Caesar, 26-letter key for Monoalphabetic).
- Client-side validation for inputs (e.g., ensuring Monoalphabetic key has 26 unique letters).
- Hover effects (e.g., logo rotation) and animations for enhanced UX.

2.4. Accessibility

- Clear error messages for invalid inputs.
- Mobile-friendly layout with adjusted logo and icon sizes.

3. System Requirements

- **Browser**: Modern browsers (Chrome, Firefox, Safari, Edge).
- **Backend**: Python 3.8+ with required packages.
- **Internet**: Optional for external resources (e.g., Font Awesome, Google Fonts).

4. Usage

4.1. Accessing the Dashboard

- Open `http://localhost:8000` after running the application.
- The dashboard loads with the **Tool** tab active.

4.2. Encrypting/Decrypting Text

1. **Select Operation**: Choose **Encrypt** or **Decrypt** from the dropdown.
2. **Select Algorithm**: Pick one of the six ciphers.
3. **Enter Text**: Input the text to process in the textarea.
4. **Enter Parameter**: Provide the required key (e.g., `3` for Caesar, `DKVQFIBJWPESCXHTMYAUOLRGNZ` for Monoalphabetic).
5. **Optional**: Check **Show step-by-step explanation** for a detailed breakdown.
6. **Submit**: Click the **Submit** button to process the text.
7. **Save Result**: Click **Save Result** to download the output as `ciphercraft_result.txt`.

4.3. Exploring Examples.

- Select an example (e.g., Monoalphabetic Cipher with text `if we wish to buy a car`).
- Click **Copy** to auto-fill the form with the example's text and key.

4.4. Switching Tabs

- **Tool**: Main interface for encryption/decryption.
- **Tutorial**: Explains how each algorithm works.
- **Guide**: Provides step-by-step usage instructions and external resources.

4.5. Toggling Dark Mode

- Click the **Dark Mode** icon (moon/sun) in the top-right header.
- The mode persists across sessions via localStorage.

5. Technical Details

5.1. Frontend

- **HTML**: `index.html` defines the structure with tabs, forms, and an examples panel.
- **CSS**: `style.css` uses Poppins font, responsive design, and animations (e.g., logo rotation, fade-in effects).
- **JavaScript**: `script.js` handles:
    - Dynamic parameter fields based on algorithm.
    - Client-side validation (e.g., Monoalphabetic key must be 26 unique letters).
    - Tab navigation, form submission via Fetch API, and result saving.
    - Examples panel toggle and copy functionality.

5.2. Backend

- **Framework**: FastAPI (`main.py`) for serving the app and processing requests.
- **Endpoints**:
    - `GET /`: Serves `index.html`.
    - `GET /documentation`: Placeholder for documentation page.
    - `POST /process`: Handles encryption/decryption with parameters (`operation`, `algorithm`, `text`, `param`).
- **Ciphers**: Implemented in Python with input validation and error handling.
    - Example: Monoalphabetic Cipher maps each letter using a 26-letter key and preserves case in output.

5.3. Dependencies

- **External**:
    - Font Awesome 6.0.0 for icons.
    - Google Fonts (Poppins) for typography.

- **Python**:
  - `fastapi`: Web framework.
  - `uvicorn`: ASGI server.
  - `jinja2`: Template rendering.

6. Example Usage

Scenario: Encrypt text using the Monoalphabetic Cipher.

1. **Inputs**:
   - Operation: Encrypt
   - Algorithm: Monoalphabetic Cipher
   - Text: `if we wish to buy a car`
   - Key: `DKVQFIBJWPESCXHTMYAUOLRGNZ`
2. **Steps**:
   - Select **Encrypt** and **Monoalphabetic Cipher**.
   - Enter the text and key.
   - Click **Submit**.
3. **Output**: `WIGFGWJOHKLZDVDA`
4. **Optional**: Enable **Show step-by-step explanation** to see the process or click **Save Result** to download the output.

7. Resources

- **Cryptography Basics**:
  - Wikipedia: Cryptography
  - Khan Academy: Cryptography
- **FastAPI Documentation**: fastapi.tiangolo.com
- **Font Awesome**: fontawesome.com
- **Google Fonts**: fonts.google.com